INFORMATION SYSTEM FOR MANAGEMENT AND ANALYSIS OF MEDICAL DATA

J. Baláš[1)], J. Rusz[1,2)], R. Čmejla[1)]

[1)] Czech Technical University in Prague, Faculty of Electrical Engineering
[2)] Charles University in Prague, First Faculty of Medicine

Abstract

This paper presents the current state of information system for management and analysis of medical data. The main aim of this project was to develop a web based system that could be used to store and easily manage large amount of data and that could provide access to this data for multiple users at the same time. The final system consists of ASP.NET application that forms the user interface and provides connection to SQL and file server. The system provides extensive user management and user access control, to suite variable needs of both small and large projects. It also contains a sophisticated search engine that makes management of large quantities of information easier. Finally, there is a basic analytical tool with potential for further improvement.

1 Introduction

As medical science advances, the number of collected medical information sharply increases. With the large amount of information, there arises the problem of how to collect, store and manage these data. In this paper, we want to introduce an information system (IS) that can help to solve this situation. Our IS is a web based application designed to ease filing and storing of large quantities of data for work-teams and individuals, and is also able to support cooperation between its users.

When working on a project or doing a long term research, it is important to keep order in files containing data.

However, this is often not respected. The most common problem is the non-existence of any rules concerning storage of specific types of data, with all participants placing their files on one "heap" on a shared network-attached storage device (NAS). Even when some rules exist, it may be difficult to explain them to the members of the team and to make everyone respect them. Even in cases where rules exist and are followed, they may not be appropriate. For example in large-scale projects, it may be difficult to find desired data when stored files do not contain classifying information according to which a search could be made (e.g. "Find all audio recordings of male patients with age between 50 and 55 who have not been diagnosed with Parkinson's disease in family").

Another problem when dealing with large amounts of data is that management of a NAS requires extensive knowledge of operation systems and networking. Further, to access NAS effectively, users usually have to install special software to operate with remote directory, such as FTP or VPN clients.

With our solution we try to overcome all these difficulties and provide a system as simple and user-friendly as possible.

2 Structure of the system

There are two points of view from which we can look on the structure of the system – physical and logical structure.

Physically the system consists of three separate parts (although they can run on the same piece of hardware)

- Web server (IIS 7.5) [3,6]

- SQL server (MS SQL server 2008) [1,2,5]
- File server (Windows server 2008)

Web server communicates directly with client's web browsers and processes their requests. Further, it provides HTML and JavaScript output through which the user communicates with the system. It runs dynamic web pages built in ASP.NET [3, 6] . It also runs code that assures data storage and retrieval on the file server, access to the data in a database and which provides user authentication and authorization in the whole system. Only the web server has access to SQL and File servers, thus detaching users and possible attackers from the stored data and at the same time providing control access to authorized users. Stored files are not placed in web page's root directory; therefore they are accessible only through scripts that are run by the web server. To prevent unauthorized users from accessing sensitive files, separate Application Pools are used for authenticated and unauthenticated users.

SQL server is used to store information about all system entities and relations between them, about users' authentication and authorization and about physical location of the stored files. Access to the SQL server is limited only to the stored procedures [1,2] that can manipulate stored data. Stored procedures use only transactions to modify the database, therefore assuring consistency of the database. Use of stored procedures also prevents malicious attacks such as SQL injection [1, 4, 7].

File server provides storage space for all uploaded files. Its structure and placement is determined by a configuration file. If file server and web server are not placed on the same hardware, web server must have access to it through Microsoft Windows network file system.

As far as the logical structure is concerned,, the system consists of variable number of applications (see fig. 1). Each of these applications is autonomous and has its own settings and root folder on the file server. Every application consists of three entity groups – patients, control patients and equipment. Each of these groups can have unlimited number of members. Every group can be defined by a series of properties. These properties are fully configurable as for the property type and user control that represents the property (check box, drop down list or text field with appropriate restrictions for defined input type – number, text etc.). Every property can be set as "required", which forces the user to fill in this property when a new member of the given group is created or updated (for example when a new patient is created and property "gender" in patient group is set as required, user cannot save this new patient until its gender is filled). For easier management and orientation, properties can be grouped into property groups. Property groups are then visibly separated in the user interface.

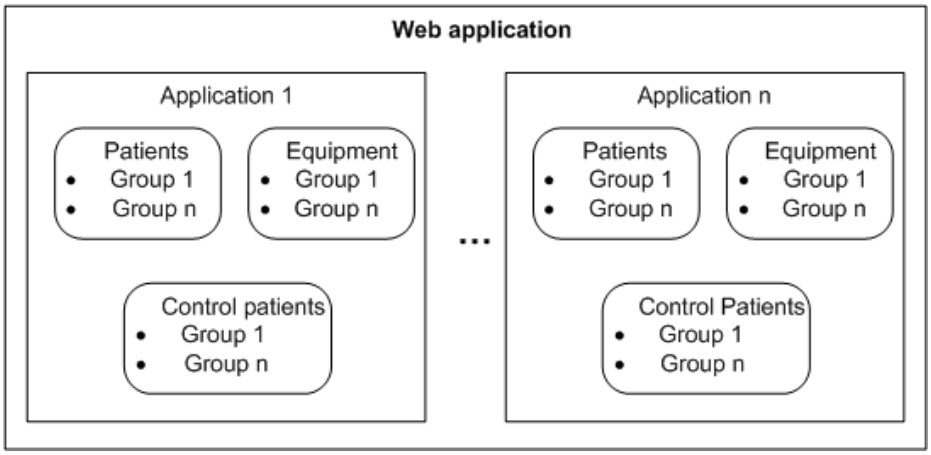

Figure 1**:** Logical structure of the application

Every member of a group can have multiple files assigned. Every file uploaded to the system is assigned to one member of one entity group.

3 Data organization and filtering

As mentioned above, every member of entity group is defined by a set of properties and can have multiple files assigned. These properties are used not only for storing informations about patients, but act as kind of meta-data for stored files. Every uploaded file assigned to a group member is defined by its name and all properties of the group member to which it is assigned. This makes categorization of multiple files much easier, since all their properties are inherited from the group member.

Moreover, these inherited properties can be used during file search. Using search engine, multiple conditions connected with logical operators can be entered. Search condition depends on property type to which it is linked (numerical property can be compared with operators <>=, verbal properties can be queries using regular expressions). This sophisticated search engine makes management of large amount of data much easier.

4 User management and access control

User management is accomplished thru the web interface. Users can be created, updated and deleted using a simple web form application. All changes are subsequently written to the database. User credentials stored in the database are used for authentication when attempts to log into the information system. Web interface is also used to set up user rights in the applications. Rights are set up both globally, for the whole information system, locally for each application and inside each application individually for every group member, test and uploaded file. Rights determine how the user interface will look like and what operations will user be able to perform on the system.

To ease the rights management, the concept of roles has been used. Role can have the same set of rights as any user, but role itself cannot be used to access the system. Instead, multiple users can be assigned to a single role and thus gain all the rights of the specific role (see fig. 2).

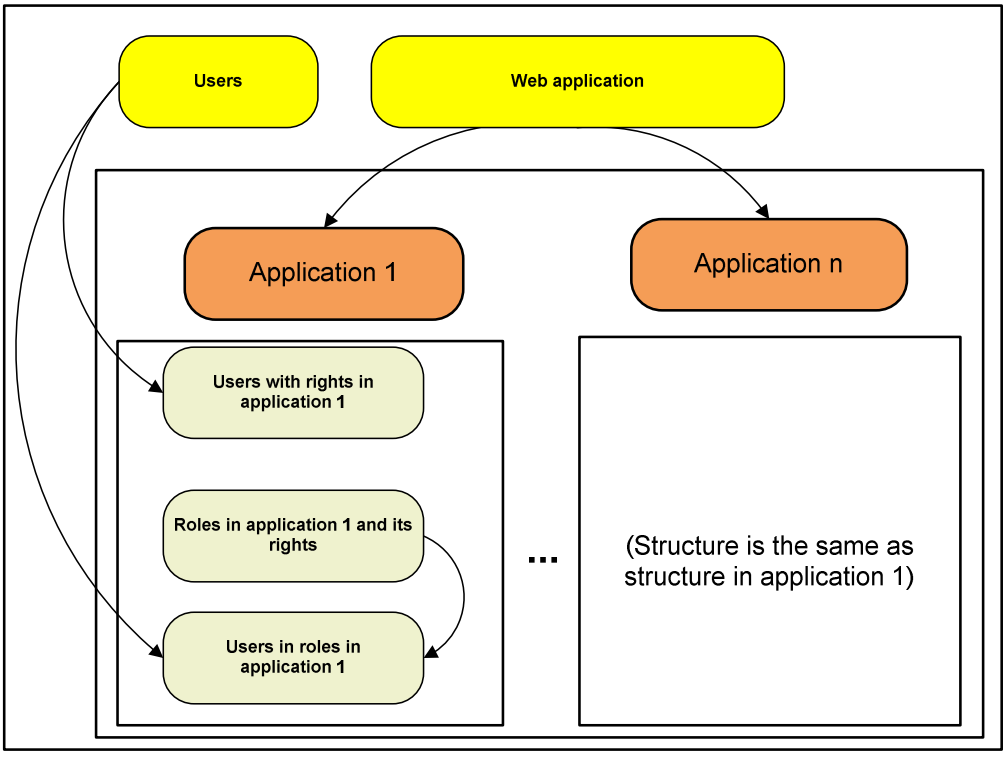

Figure 2: Application's rights structure

5 Test and analysis

The system contains environment for on-line testing of uploaded audio files. Audio recordings are evaluated by skilled professionals and recorded values are stored in the database for further processing or analysis in the system.

Test-creator can select multiple uploaded audio files and create a test question which will be answered by multiple users (professionals) for every selected audio file. Audio files can be played during the test directly from user's web browser without the necessity to download each file. Answers to the question are always a number.

Analytical tool allows seeing answer ranges for selected tests in gant graph. It is quite useful to determine usefulness of acquired data. Similar analysis with pie of gant graph output can be done with any number of patients or control patients with numerical properties.

6 Conclusions and future plans

This paper presents information system for management and analysis of medical data. The developed system is fully capable of storing and managing a large amount of various types of data and provides usable user interface for controlling the whole system. Various biological signals uploaded on the server are accessible through the system and can be easily managed by students. Using the MATLAB environment, students have subsequently the opportunity to analyse and design basic scripts performed on their own or other available signals.

In the future, development will focus mainly on development of multiple import based on XML input file to allow faster import of large amount of data. We further plan to implement connection to Matlab or similar mathematical tool to allow execution of custom user scripts over the stored files and thus extend analytical possibilities of the system.

Acknowledgement

This work has been primarily supported by the grants SGS10/180/OHK3/2T/13 Voice and Speech Disorders Evaluation, and GAČR102/08/H008 Analysis and modelling biomedical and speech signals. Study has been also supported by the research programs MSM6840770012 Transdisciplinary Research in Biomedical Engineering, and MSM 0021620806 Studies at the molecular and cellular levels in normal and in selected clinically relevant pathologic states.

References

[1] Brust, Andrew J. a Forte, Stephen. 2007. *Mistrovství v programování SQL Serveru 2005*. Brno : Xomputer Press, a.s., 2007. 978-80-251-1607-4.

[2] Dewson, Robin. 2008. Beginning SQL Server 2008 for Developers: From Novice to Professional. New York : Apress, 2008. 978-1-59059-958-7.

[3] Duckett, Jon. 2004. *Beginning Web Programming with HTML, XHTML, and CSS.,* Wiley Publishing, Inc., 2004.

[4] Hernandez, Michael J. a Viescas, John L. 2004. *Myslíme v jazyku SQL.* Praha : GRADA, 2004. 80-247-0899-X.

[5] Kellenberger, Kathi. 2009. *Beginning T-SQL 2008.* New York : Apress, 2009. 978-1-4302-2461-7.

[6] MacDonald, Matthew. 2008. Beginning ASP.NET 3.5 in C# 2008: From Novice to Professional, Second Edition. New York : Apress, 2008. 978-1-59059-891-7.

[7] Pokorný, Jaroslav a HALAŠKA, I. 1998. *Databázové systémy.* Praha : ČVUT, 1998. 80-01-01724-9.

Jan Baláš
Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Circuit Theory, Czech Republic
e-mail: balasjan@fel.cvut.cz

Jan Rusz
Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Circuit Theory, Czech Republic
Charles University in Prague, First Faculty of Medicine, Department of Pathological Physiology, Laboratory of Biocybernetics, Czech Republic
e-mail: ruszjan@fel.cvut.cz

Roman Čmejla
Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Circuit Theory, Czech Republic
e-mail: cmejla@fel.cvut.cz